

U.S. DEPARTMENT OF COMMERCE
PATENT AND TRADEMARK OFFICE

PATENT APPLICATION TRANSMITTAL LETTER

ATTORNEY DOCKET NUMBER:
2685/113031

Commissioner of Patents and Trademarks
Washington D.C. 20231

Transmitted herewith for filing is the patent application of

Inventor(s): **Steven M. Bellovin**

For : **METHOD AND APPARATUS FOR UPDATING LISTS FOR A DISTRIBUTED FIREWALL**

Enclosed are:

1. **24** sheets of specification, **6** sheet(s) of claims, **1** sheet of abstract, and **3** sheet(s) of drawings;

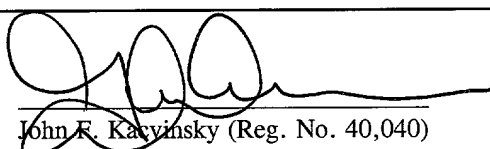
The filing fee has been calculated as shown below:

	NUMBER FILED	NUMBER EXTRA*	RATE (\$)	FEE (\$)
BASIC FEE			760.00	760.00
TOTAL CLAIMS	29 - 20 =	9	18.00	162.00
INDEPENDENT CLAIMS	4 - 3 =	1	78.00	78.00
MULTIPLE DEPENDENT CLAIM PRESENT				1,000.00
FEE FOR RECORDATION OF ASSIGNMENT			0.00	.00
*Number extra must be zero or larger			TOTAL	\$1,000.00
If applicant is a small entity under 37 C.F.R. §§ 1.9 and 1.27, then divide total fee by 2, and enter amount here.			SMALL ENTITY TOTAL	.00

4. The Office is authorized to charge the filing fee of **\$1,000.00** to **Deposit Account No. 11-0600**. A duplicate copy of this paper is enclosed for that purpose.

5. Please direct all correspondence to **Frank Pietrantonio** at **Kenyon & Kenyon, 1500 K Street, N.W., Suite 700, Washington, DC 20005**, phone number **(202) 220-4200**.

Dated: June 30, 1999


John F. Kacvinsky (Reg. No. 40,040)

KENYON & KENYON
1500 K Street, N.W.
Suite 700
Washington, DC 20005
Phone: (202) 220-4200
Fax: (202) 220-4201

15525 U.S. PTO
09/31/99



A METHOD AND APPARATUS FOR A DISTRIBUTED FIREWALL

FIELD OF THE INVENTION

5 The invention relates to communications in general. More particularly, the invention relates to a method and apparatus for creating a distributed firewall using cryptographic techniques.

BACKGROUND OF THE INVENTION

10

Conventional firewalls rely on the notions of restricted topology and control entry points to function. More precisely, they rely on the assumption that everyone on one side of the firewall is to be trusted, while those on the other side are not. The vastly expanded Internet connectivity in recent years has called that assumption into
15 question. So-called “extranets” can allow outsiders to reach the “inside” of the firewall. Conversely, telecommuters’ machines that traditionally do not have the protection of a firewall use the Internet for connectivity. These machines in particular need protection when encrypted tunnels are not in place. Other trends are also threatening the traditional notion of firewalls. For example, some machines need

more access to the outside than do others. Conventional firewalls are not suited to deal with these types of problems, especially as internally assigned Internet Protocol (IP) addresses change. End-to-end encryption is another threat, since the firewall generally does not have the necessary codes (or keys) to peer through the encryption.

5 In an attempt to solve some of these problems, the notion of a “distributed firewall” was created. Instead of having a single firewall existing at some controlled access point to the network, each network device would be equipped with its own firewall. A distributed firewall preserves central control of access policy, while reducing or eliminating any dependency on topology. In such a scheme, policy is still
10 centrally defined. Enforcement of the policy, however, takes place on each endpoint. In this manner the advantages of conventional firewalls are maintained while avoiding most of the associated limitations, most notably the dependency on topology.

One example of a distributed firewall is described in United States Patent Number 5,606,668 titled “System for Securing Inbound and Outbound Data Packet
15 Flow in a Computer Network.” The ‘668 patent discloses a distributed firewall wherein each network device is equipped with its own packet filter. A packet filter simply retrieves a source address from a packet, and compares it with a list of addresses. The packet is then passed or dropped based on the results of the comparison. The determination whether to pass or drop the packet is normally
20 implemented as a set of rules, e.g., if the address is on the list drop the packet.

Conventional distributed firewalls, such as the one disclosed in the ‘668 patent, however, are less than satisfactory for a number of reasons. For example, the distributed firewall described in the ‘668 patent is not capable of preventing “spoofing.” Spoofing refers to a technique wherein a packet sender attempts to access
25 a firewall by impersonating another sender’s address. For example, the filter module described in the ‘668 patent filters packets based on the source IP address embedded within the packet. A person could discover a source IP address with permission to

access a secure site, and attempt to access the secure site by emulating the allowed source IP address. In other words, there is no technique for authenticating the identity of the sender. In another example, the '668 patent fails to address the telecommuting situation where a user often connects a computer to a corporate network via an unsecured link, e.g., the Internet. In such a case, there is no mechanism for protecting communications between the corporate network and the telecommuters' computer. Part of the problem with telecommuters is triangle routing, plus lack of protection for their machines when on the outside Internet.

From the foregoing, it can be appreciated that a substantial need exists for a distributed firewall that solves the aforementioned problems.

SUMMARY OF THE INVENTION

An embodiment of the invention comprises a method and apparatus for a implementing a distributed firewall. A packet filter processor receives a packet sent from a first device to a second device. The packet filter processor authenticates an identifier for the packet. For example, authentication could be performed using a cryptographically-verifiable identifier. The packet filter processor determines whether to send the packet to the second device, based on the authentication and a set of policy rules. The packet filter processor sends the packet to the second device in accordance with the determination.

With these and other advantages and features of the invention that will become hereinafter apparent, the nature of the invention may be more clearly understood by reference to the following detailed description of the invention, the appended claims and to the several drawings attached herein.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 illustrates a network topology suitable for practicing one embodiment of the invention.

5 FIG. 2 is a block diagram of a packet filter processor in accordance with one embodiment of the invention.

FIG. 3 is a block flow diagram of steps for filtering data packets in accordance with one embodiment of the invention.

10 **DETAILED DESCRIPTION**

The embodiments of the invention include a distributed firewall that utilizes the following principles: a) a policy language that states what sort of connections are permitted or prohibited, b) an encryption mechanism; and c) any of a number of
15 system management tools, such as Microsoft's Systems Management Server (SMS) or Automatic Software Distribution (ASD) to manage the distributed firewall. ASD is described in a paper authored by Andrew Koenig entitled "Automatic Software Distribution," USENIX Conference Proceedings, Pages 312-322, Salt Lake City, UT, Summer, 1984, and is incorporated by reference herein.

20 In one embodiment of the invention, a compiler translates the policy language into some internal format. The system management software distributes this policy file to all hosts that are protected by the firewall. Any incoming packets are accepted or rejected by each "inside" host, according to both the policy and the cryptographically-verified identity of each sender.

25 The embodiments of the invention, unlike conventional firewalls, incorporate an encryption mechanism used to authenticate the cryptographically-verified identity of each sender to add another layer of protection against unauthorized intrusions.

Some conventional packet filters evaluate an incoming packet using a source designator such as an IP address (e.g., the source address). That is, a specified IP address may be fully trusted, able to receive incoming mail from the Internet, and so forth. Other conventional packet filters rely on topology to know the difference

5 between "inside" and "outside", where packets from "outside" are only allowed to reach certain machines and services on the "inside". As stated above, however, the specified IP address may be duplicated by an unauthorized party or agent and used to gain access to a secure site. To avoid this problem, the embodiments of the invention utilize a name in a cryptographic certificate as the common host identifier.

- 10 Certificates can be a very reliable unique identifier for many reasons. For example, they are independent of topology, and therefore the use of certificates removes the "inside" and "outside" distinction. Furthermore, ownership of a certificate is not easily spoofed. If a machine is granted certain privileges based on its certificate, those privileges can apply regardless of where the machine is located physically. In a
- 15 different sense, policies can be "pulled" dynamically by the end system. For example, a license server or a security clearance server can be asked if a certain communication should be permitted.

The combination of a distributed firewall and cryptographically-verified identity authentication provides a superior form of protection over conventional

20 firewalls for a number of reasons. As an initial matter, distributed firewalls in general offer many advantages over conventional firewalls. For example, a distributed firewall can be centrally managed. In a typical organizational environment, individuals are not necessarily the administrators of the computers they use. Instead, to simplify system administration and to permit some level of central control, a system

25 management package is used to administer individual machines. Patches can be installed, new software distributed, and so forth. These same mechanisms are used to control a distributed firewall. Policy is enforced by each individual host that

participates in a distributed firewall. The security administrator, who is no longer necessarily the "local" administrator, defines the security policy in terms of host identifiers. The resulting policy (compiled to some convenient internal format) is then shipped out, much like any other change. This policy file is consulted before

5 processing incoming or outgoing messages, to verify their compliance. It is most natural to think of this happening at the network or transport layers, but policies and enforcement can equally well apply to the application layer, data link layer or physical layer, if desired. For example, some sites might wish to force local Web browsers to disable Java or Javascript.

10 Policy enforcement is especially useful if the peer host is identified by a certificate. If so, the local host has a much stronger assurance of its identity than in a traditional firewall. In the latter case, all hosts on the inside are in some sense equal. If any such machines are subverted, they can launch attacks on hosts that they would not normally talk to, possibly by impersonating trusted hosts for protocols such as
15 "rlogin." With a distributed firewall, though, such spoofing is not possible since each host's identity is cryptographically assured.

Consider the problem of electronic mail. Because of a long-standing history of security problems in mailers, most sites with firewalls let only a few, designated hosts receive mail from the outside. They in turn will relay the mail to internal mail servers.

20 Traditional firewalls would express this by a rule that permitted Simple Mail Transfer Protocol (SMTP) (port 25) connections to the internal mail gateways; access to other internal hosts would be blocked. On the inside of the firewall, though, access to port 25 is unrestricted. With a distributed firewall, all machines have some rule concerning port 25. The mail gateway permits anyone to connect to that port. Other
25 internal machines, however, permit contact only from the mail gateway, as identified by its certificate. This provides superior protection than conventional techniques.

For example, even a subverted internal host cannot exploit possible mailer bugs on the

protected machines.

Distributed firewalls have other advantages as well. The most obvious is that there is no longer a single choke point. From both a performance and availability stand-point, this is a major benefit. Throughput is no longer limited by the speed of the firewall. Similarly, there is no longer a single point of failure that can isolate an entire network. A second advantage is more subtle. Conventional firewalls do not have certain knowledge of what a host intends. Instead, they have to rely on externally-visible features of assorted protocols. Thus, an incoming TCP packet is sometimes presumed legitimate if it has the "ACK" bit set, since such a packet can only be legitimate if it is part of an ongoing conversation (e.g., a conversation whose initiation was presumably allowed by the firewall). But spoofed ACK packets can be used as part of "stealth scanning." Similarly, it is hard for firewalls to treat User Datagram Protocol (UDP) packets properly, because they cannot tell if they are replies to outbound queries, and hence legal, or if they are incoming attacks. The sending host, however, knows. Relying on the host to make the appropriate decision is therefore more secure. This advantage is even clearer when it comes to protocols such as the File Transfer Protocol (FTP). By default, FTP clients use the PORT command to specify the port number used for the data channel. This port is for an incoming call that should be permitted, an operation that is generally not permitted through a firewall. Conventional firewalls, even the stateful packet filters, generally use an application-level gateway to handle such commands. With a distributed firewall, the host itself knows when it is listening for a particular data connection, and can reject random probes. One particular advantage of distributed firewalls, however, is that distributed firewalls can protect hosts that are not within a topological boundary.

Consider a telecommuter who uses the Internet both generically and to tunnel in to a corporate net. A problem arises regarding how the telecommuter's computer

should be protected. A conventional approach can protect the machine using conventional “tunneling” techniques, but that requires that generic Internet use be tunneled into the corporate network and then back out the Internet. Apart from efficiency considerations, such use is often in violation of corporate guidelines.

5 Furthermore, there is no protection whatsoever when the tunnel is not set up. By contrast, a distributed firewall protects the machine all of the time, regardless of whether or not a tunnel is set up. Corporate packets, authenticated by an encryption mechanism, are granted more privileges, while packets from random Internet hosts can be rejected. This would also avoid the need for triangle routing.

10 The embodiments of the invention utilize a policy language that states what sort of connections are permitted or prohibited. Many possible policy languages can be used as long as the particular policy language is powerful enough to express the desired policy or rule set. In this embodiment of the invention, the general policy language provided by KeyNote is used. KeyNote is an especially attractive choice for
 15 a policy language due to its suitability for configuring packet filters. Its advantages include the integration of credentials with policy specification, and an ability to use a single mechanism to specify policy at different levels. KeyNote is described in a paper authored by M. Blaze, J. Feigenbaum and A. Keromytis, entitled “KeyNote: Trust Management for Public-Key Infrastructures,” Springer, LNCS vol. 1550,
 20 Proceedings of the 1998 Cambridge Security Protocols International Workshop, 1999, pages 59-63, which is incorporated by reference herein.

The policy rules are designed by a trusted authority, such as a site administrator, and are stored in a policy configuration file. A sample policy configuration file is shown as follows:

25

```
{Start}
    inside_net = x509{name="*.example.com"};
```

```
mail_gw = x509{name="mailgw.example.com"};
```

```
time_server = IPv4{10.1.2.3};
```

```
allow smtp(*, mail_gw);
```

```
5 allow smtp(mail_gw, inside_net);
```

```
allow ntp(time_server, inside_net);
```

```
allow *(inside_net, *).
```

```
{End}
```

10 According to this sample policy configuration file, electronic mail from outside the distributed firewall can only reach the machine with a certificate identifying it as the mail gateway. The mail gateway then sends the received mail to all “inside” machines. The terms “inside” and “outside” as used herein refer to those devices or applications that are protected by the distributed firewall, and those devices or

15 applications that are seeking entrance to the protected devices or applications, respectively. This sample policy configuration file also makes reference to the Network Time Protocol (NTP), which is a low-risk protocol that has its own application-level protection. Any NTP messages are allowed to be distributed from a given IP address to all inside machines. Finally, all outgoing calls from inside devices

20 to outside devices are permitted.

In addition to the policy language and system management tools, the embodiments of the invention utilize an encryption mechanism to strengthen the security offered by conventional distributed firewalls. The encryption mechanism can be any cryptographic technique which is resistant to decoding by an untrusted party.

25 In one embodiment of the invention, the encryption mechanism used is the one set forth in the Internet Engineering Task Force’s Request For Comments 2401, entitled a “Security Architecture For The Internet Protocol”, authored by S. Kent and R.

Atkinson, November 1998 (IPSEC), which is incorporated by reference herein.

IPSEC is a network-level encryption mechanism for the Transmission Control Protocol/Internet Protocol (TCP/IP). In accordance with IPSEC, each packet can be modified to include an Authentication Header (AH). The AH provides connectionless integrity, data origin authentication, and an optional anti-replay service. The AH is a vehicle for access control, based on the distribution of cryptographic keys and the management of traffic flows relative to these security protocols. The details and implementation of the AH is described in detail in IPSEC, and will not be further described herein.

10 The encryption mechanism can be used in a number of different ways to enhance the functionality provided by distributed firewalls. In one particularly advantageous embodiment, end-to-end IPSEC is used. Each incoming packet can be associated with a certificate. The access granted to that packet is determined by the rights granted to that certificate. Consider a packet destined for port 25 (SMTP) on
15 an inside machine, given the sample policy configuration file described above. If the certificate identifies the source as "mailgw.example.com" (note that this is not necessarily the same as the machine with that domain name) the packet will be accepted. If the certificate name is different (for clarity, any discussion of the certificate's signature chain is omitted, although it is clearly important for real use),
20 or if there is no IPSEC protection, the packet will be dropped as unauthorized. It is noted that the necessary filtering is prescribed by the IPSEC architecture. Specifically, the inbound Security Policy Database (SPD) is used to reject illegal input packets, while the outbound SPD can be used to control outgoing connections. Many commercial IPSEC implementations either support port number-granularity security
25 associations or will in the near future. Application-level protection can be achieved by distributing application-specific policy files. Thus, web browsers can be told, by the central site, to reject, for example, all ActiveX controls or Java applets. This is a

particularly difficult function for conventional firewalls. Doing it on the end hosts is more secure, if the policy distribution problem can be solved. One of the more difficult problems is handling protocols such as FTP without touching the application. This can be accomplished with per-process keying for IPSEC. For example, a policy rule for FTP would indicate that outbound connections to port 21 must be protected with IPSEC, and that all other TCP connections protected by that security association (SA) are legal. Since only that process can use that SA, and it would only have the FTP data channel open, an adequate level of protection can be achieved.

Given that access rights in a strong distributed firewall are tied to certificates, access rights can be limited by changing the set of certificates accepted. Suppose, for example, that a security flaw is discovered in some networked application. A new certificate can be distributed to hosts in the same distribution as the patch. Only hosts with newer certificates are then considered to be "inside." If the change is not installed, the machine will have fewer privileges. In some environments, it may even be possible to use certificates to help ensure user compliance. A "PolicyMaker" certificate could contain code that checks the configuration of the user's system. A PolicyMaker certificate is described in a paper authored by Matt Blaze et al. entitled "Decentralized Trust Management," IEEE Symposium on Security and Privacy, pages 164-173, 1996, and a paper authored by Matt Blaze et al. entitled "Compliance Checking In The PolicyMaker Trust Management System," Proceedings of the 2nd Financial Crypto Conference, 1998, both of which are incorporated by reference herein. If the check failed, that is, if the proper versions of the proper files were not present, the certificate would not allow itself to be used for authentication. Without tamper-resistant hardware this is not an absolute check, but it can help protect against accidental misconfiguration and less-determined malfeasance. The certificate version mechanism also protects against new, insecure machines that are installed on an inside network. Until the appropriate filtering software and rule sets are installed (and

possibly until the machine is otherwise made secure by the organization's administrators), no certificate is issued. The machine is thus an outside machine, regardless of its physical location.

Referring now in detail to the drawings wherein like parts are designated by like reference numerals throughout, there is illustrated in FIG. 1 a network topology suitable for practicing one embodiment of the invention. As shown in FIG. 1, a router 104 is connected to a network 102. Router 104 is in turn connected to a gateway 106. Gateway 106 is connected to a local area network (LAN) 112 having one or more work stations (W/S) 108. Each workstation 108 is connected to LAN 112 by a packet filter processor 110. A workstation can be any type of network device, such as a personal computer, network server, laptop computer, personal digital assistant (PDA), and so forth. Furthermore, the workstation can be connected directly to LAN 112, or indirectly via network 102. In addition, any one of the workstations can be used as a central site administrator. This workstation would be configured with the appropriate policy language and system management software/hardware (collectively referred to as a "system management module") to manage each packet filter processor 110 in accordance with the principles described herein.

In this embodiment of the invention a dedicated packet filter processor is used to help enhance the speed and throughput of the packet filter. By adding additional overhead to each packet in the form of an IPSEC authentication header, encrypting the entire packet itself, or a combination of both, the processing of packets by the packet filter becomes incrementally more burdensome. Therefore, it is useful to have a dedicated packet filter to offset the increased processing requirements. Although this embodiment of the invention use dedicated packet filter processors for the distributed firewall, however, software based filter modules could also be used and still fall within the scope of the invention. An example of such software based filter modules

is described in the '668 patent, modified to reflect the principles detailed herein.

Further, packet filter processor 110 can be placed anywhere in the network such as, for example, gateway 106, router 104 or a network device connecting from outside the network via the Internet, or in any other strategic point inside or outside the

5 network, and still fall within the scope of the invention. Finally, it can be appreciated that the particular configuration shown in FIG. 1 is chosen as an example only and is not limitive of the type of network on which the embodiments of the present invention can work. The number of configurations that networks can take are virtually limitless and techniques for setting up those configurations are well known to those skilled in
10 the art. The embodiments of the invention can operate on any of these possible configurations.

In the embodiments of the invention, network 102 and LAN 112 are packet based networks, such as Transmission Control Protocol/Internet Protocol (TCP/IP) networks or X.25 networks. A packet originates from network 102 or LAN 112 with
15 an intended destination to workstation 108. Both the source and destination addresses are included in the packet, as well a cryptographically-verifiable identifier for the host or sending device. In one embodiment of the invention, the cryptographically-verifiable identifier is part of an AH in accordance with IPSEC.

FIG. 2 is a block diagram of a packet filter processor in accordance with one
20 embodiment of the invention. As shown in FIG. 2, LAN interface (I/F) connector 20 is coupled to network interface card 22. Connector 20 and card 22 are used to interface with LAN 112, and to accept packets originating from network 102 or workstation 108. Connector 48 and card 46 are used to interface with additional networks or network devices if a packet filter processor is placed in an
25 internetworking device such as a router, bridge or gateway. Connectors 20 and 48, as well as cards 22 and 46, operate in accordance with principles well-known in the art.

Further, cards 22 and 46 are designed to adhere to the Institute of Electrical

and Electronics Engineers (IEEE) standard titled "Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications, American National Standard ANSI/IEEE Standard 802.3, 1985 ("IEEE 802.3 standard"). The IEEE 802.3 standard defines a technique referred to as CSMA/CD, which is appropriate for a network having a bus/tree topology. It can be appreciated, however, that network interfaces designed to work with other medium access techniques or standards could be used for packet filter processor 110, and still fall within the scope of the invention.

Cards 22 and 44 are connected to one another, and also to First In First Out (FIFO) buffers 24 and 44, respectively. FIFO buffers 24 and 44 are used to store incoming or outgoing packets in memory until each packet can be compared and sent to appropriate network.

Packet filter processor 110 also includes several types of high-speed memory. By way of example, this embodiment of the invention includes a 96 kilobyte (K) Programmable Read Only Memory (PROM) 40, a 32K Non-Volatile Random Access Memory (NVRAM) 42, and a Dynamic Random Access Memory (DRAM) bank 32. There is also a DRAM control 30 for DRAM bank 32.

Each type of memory is used to store data for packet filter processor 110. For example, PROM 40 is used to store an operating system 39 (not shown) for packet filter processor 110. NVRAM 42 is used to store user defined parameters 45, and operating system parameters 43 used by the operating system stored in PROM 40. DRAM bank 32 is used to store a common host identifier list 33 of common host identifiers. DRAM bank 32 also stores a IPSEC module and its associated databases, such as the Security Policy Database (SPD), the Security Association Database (SAD), and a policy configuration file. The SPD specifies the policies that determine the disposition of all IP traffic inbound or outbound from a host, security gateway, and so forth. The SAD contains parameters that are associated with each active SA.

The policy configuration file is designed using a particular policy language, and represents the policy or rules by which packets are processed by packet filter processor 110. Alternatively, the IPSEC module and its associated databases (SAD and SPD) and policy configuration file could be stored in memory within the network
5 device where packet filter processor 110 is implemented (e.g., a PC's RAM).

The heart of packet filter processor 38 is a dedicated high performance microprocessor 38. Any microprocessor capable of operating at the speeds necessary to implement of the functions of the packet filter processor is appropriate. Examples of processors suitable to practice the invention includes the INTEL family of
10 processors, such as the Pentium®, Pentium® Pro, and Pentium® II microprocessors.

Packet filter processor 110 also includes a connector 34 and interface 36, both of which are attached to processor 38. Connector 34 and interface 36 both adhere to Electronic Industries Association (EIA) Standard RS-232-C titled "Interface Between Data Terminal Equipment and Data Communication Equipment Employing Serial
15 Binary Data Interexchange," October, 1969. Finally, packet filter processor 110 includes a clock 26 and clock counter 28 to control the timing of packet filter processor 110.

Packet filter processor 110 operates in accordance with operating system 39, the IPSEC module and the policy configuration file, each of which is comprised of a
20 set of computer program instructions which are stored in PROM 40. Since a list of common host identifiers can include a large number of identifiers, e.g., ranging from hundreds to several thousand, the processing time required to compare an identifier of an incoming packet with a list of several thousand identifiers is enormous, and significantly degrades the performance of many conventional packet filters. By way
25 of contrast, packet filter processor 110 combines the elements of a high-speed microprocessor, a common host identifier list stored in high-speed memory, and a dedicated proprietary operating system, to ensure that data packets can be filtered at a

high-rate of speed.

Operating system 39 is designed to control the operation of the processor in conjunction with the IPSEC module. More particularly, operating system 39 and the IPSEC module are designed such that the processor is directed to look at the common host identifier of each received data packet to determine if the common host identifier matches one of the stored common host identifiers, and if there is a match, to either discard or forward the data packet depending on the rules set forth in the policy configuration file. Moreover, it verifies that the packet is successfully authenticated by a key belonging to an identified host. This lets the filter processor know from which host the packet was sent. Since operating system 39 and processor 38 are dedicated to one task, packet filter processor 110 can perform the filtering process very quickly and efficiently. The operation of operating system 39, and of packet filter processor 110 in general, will be described in more detail with reference to FIG. 3.

Another reason packet filter processor 110 is so efficient is that packet filter processor 110 is implemented between the physical layer and data link layer of the International Standardization Organization (ISO) protocol stack. The significance of this implementation can be better appreciated in view of some background information of network architectures in general.

A network architecture defines protocols, message formats, and standards to which products must conform in order to connect properly with the network. Architectures are developed by standards organizations, common carriers, and a computer and network vendors. Network architectures use a layered approach, whereby functions are organized into groups and assigned to specific functional layers in the architecture. Network architectures define the interfaces between layers in a given network node and within the same layer in two different nodes.

The ISO model provides a generalized model of system interconnection. It

encompasses seven layers: application, presentation, session, transport, network, data link, and physical. A brief summary for each layer is given as follows:

1. Physical Layer - The physical layer is responsible for the transmission of bit stream across a particular physical transmission medium. It involves a connection between two machines that allows electrical signals to be exchanged between them.
2. Data Link Layer - The data link layer is responsible for providing reliable data transmission from one node to another and for shielding higher layers from any concerns about the physical transmission medium. It is concerned with the error free transmission of frames of data.
3. Network Layer - The network layer is concerned with routing data from one network node to another. It is responsible for establishing, maintaining, and terminating the network connection between two users and for transferring data along that connection.
4. Transport Layer - The transport layer is responsible for providing data transfer between two users at an agreed on level of quality.
5. Session Layer - The session layer focuses on providing services used to organize and synchronize the dialog that takes place between users and to manage data exchange.
6. Presentation Layer - The presentation layer is responsible for the presentation of information in a way that is meaningful to the network users, e.g., character

code translation, data conversion, or data compression or expansion.

7. Application Layer - The application layer provides a means for application processes to access the system interconnection facilities in order to exchange information.

Packet filter processor 110 is implemented between the physical layer and data link layers described above, in order to increase the speed at which packets are filtered. The physical layer is responsible for data encoding and decoding. Data encoding refers to translating the bits being transmitted into the proper electrical signals to be sent across the transmission medium. Data decoding translates the electrical signals received over the transmission medium into the bit stream those signals represent. The data link layer is concerned with data encapsulation/decapsulation and media access management. These functions, however, are not necessary for identifying the common host identifier of the packet. For example, data decapsulation is the function of recognizing the destination address, determining if it matches the receiving station's address, performing error checking, and removing control information that was added by the data encapsulation function in the sending station. Therefore, by implementing packet filter processor 110 between the physical layer and data link layer, processor 110 can maximize the speed at which it filters each packet.

Although packet filter processor 110 is implemented between the physical layer and data link layer in this embodiment of the invention, it can be appreciated that packet filter processor can be implemented at or between other layers as well. For example, it is possible to implement packet filter processor 110 between the data link layer and network layer, or at the network layer itself, and still fall within the scope of the invention.

FIG. 3 illustrates a block flow diagram of steps for filtering data packets in accordance with one embodiment of the invention. The description with respect to FIG. 3 will assume that a packet is originating from network 102 and has an intended destination address of workstation 108. It can be appreciated, however, that the operation of packet filter processor 110 is identical when the packet originates from workstation 108 and has an intended destination address within network 102 or another workstation attached to network 102. Furthermore, each packet is assumed to be modified to include a cryptographically-verifiable common host identifier as part of the AH in accordance with IPSEC. In one embodiment of the invention, each packet is further encrypted in its entirety to add another layer of protection.

Packet filter processor 110 receives a packet at step 50. Connector 20 receives the packet and passes the packet to interface card 22 which is designed to convert the electrical impulses received over the physical transmission media into packets conforming to the standards set forth in IEEE 802.3. The packet is stored in FIFO 24.

Processor 38 authenticates the common host identifier for the packet in accordance with IPSEC standards at step 52. A security association is set up for each source device using a certificate and stored in the SAD. The SA includes the common host identifier. When a packet is received, the AH header has a pointer to a SA stored in the SAD. The SA has (among other things) a cryptographic key. The key is used to authenticate the packet by means of a cryptographic function as set forth in IPSEC. If the check succeeds, the common host identifier is taken from the SAD as indicated by the SA. The common host identifier could be a port number, a source IP address, or some other designator. If the entire packet is encrypted, the entire packet is decrypted and the above process is performed for the AH.

In one embodiment of the invention, encryption/decryption for key exchange and certificate verification to set up a SA is performed using a public key

cryptography algorithm, such as the algorithm originally developed by Ronald Rivest, Adi Shamir and Leonard Adleman (RSA) in 1978 or any of its updates. In public key cryptography, two keys are used which are typically referred to as a “key pair.” The first key of the key pair is known as the public key and is used to encrypt messages.

- 5 The second key of the key pair is known as the corresponding private key and is used for decryption. Given only the public key, it is effectively impossible to find the corresponding private key, and hence impossible to read encrypted messages. Other encryption techniques are used for encryption and authentication of packets, such as DES, triple DES for encryption, and HMAC.

- 10 Once the common host identifier has been authenticated at step 52, the common host identifier is compared to list 33, which is stored in DRAM bank 32, at step 54. List 33 is stored in DRAM bank 32 in order to increase the speed at which data from the list could be retrieved by processor 38, as compared to, e.g., when data is stored on some other computer readable medium such as a hard drive or floppy
- 15 disk. Step 56 comprises a test to determine whether there is a match at step 54. If there is a match at step 54, then packet filter processor 58 records the attempt at step 58 before passing control to step 60. If there is not a match at step 54, then control is directly passed to step 60.

- Packet filter processor 110 determines whether the packet should be passed at
- 20 step 60. The decision whether to pass the packet or not is dependent upon the rules set forth in the policy configuration file for processor 110. For example, packet filter processor 110 could have a policy rule set indicating that processor 110 should operate in a restrictive mode or a permissive mode. Restrictive mode refers to a condition where a select number of packets are to be passed, and all others blocked.
- 25 Permissive mode is where all packets are to be passed except for a select few that require blocking. Thus, in permissive mode, the packet is passed if the common host identifier for a packet does not match an identifier on list 33. If there is a match,

packet filter processor 110 drops the packet. In restrictive mode, the packet is passed if the common host identifier does match an identifier from list 33, and is dropped otherwise. In addition, a rule could be implemented that if a common host identifier does not match an identifier from list 33, that processor 110 seeks authority from
5 another device regarding treatment of the packet.

At step 60, packet filter processor 110 determines whether the packet should be passed depending on whether processor 110 has been set to permissive mode or restrictive mode. If processor 110 has been set to restrictive mode, and there is a match at step 56, then the packet is passed at step 62 to the destination network which
10 in this embodiment of the invention is network 16 or terminal 18. If processor 110 has been set to restrictive mode, and there is not a match at step 56, then the packet is dropped at step 64. Conversely, if processor 110 has been set to permissive mode, and there is a match at step 56, then the packet is dropped at step 64. If processor
15 110 has been set to permissive mode, and there is not a match at step 56, then the packet is passed to the destination network at step 62. In this embodiment of the invention, a default condition is that no feedback is given to the system sending the packets for security reasons if a packet is dropped at step 64. It can be appreciated, however, that this default condition can be changed and still fall within the scope of the invention.

20 In accordance with the system administration aspects of the invention, a service provider administers a database of common host identifier lists. Each list may contain the common host identifiers of particular types of Internet sites. The service provider periodically updates list 33 stored in DRAM bank 32 of packet filter processor 110 in a secure manner using cryptography. In this way, end users can be
25 assured that the common host identifier lists stored in their filtering processor are up to date and that the data has not been tampered with by intercepting users.

List 33 can be updated in at least two ways. First, list 33 could be updated by

connecting Data Terminal Equipment (DTE) such as an asynchronous (ASCII) terminal (or personal computer emulating an asynchronous terminal) to RS-232 connector 34 of packet filter processor 110. This method would enhance security when updating list 33. Alternatively, a network connection is formed with a central administrative site equipped with a list server 70, preferably through an Internet Service Provider (ISP) using a direct network connection or via RS-232 connector 34. List 33 is then updated from the central administrative site, either by a request by the list server 70 of the administrative site, or on the request of packet filter processor 110.

Although a full implementation of distributed firewalls is the most secure and the most flexible, hybrid implementations can exist. That is, one can combine the techniques described here with traditional firewalls, achieving adequate functionality at lower cost, especially until IPSEC support becomes ubiquitous. In a hybrid implementation, some hosts are behind a traditional firewall, while other hosts live on the outside. An IPSEC gateway at the central site provides connectivity to the outside machines. This configuration is common at companies with a major central site and some number of telecommuters. As in ordinary virtual private networks (VPNs), remote hosts have full access to the inside, by virtue of the IPSEC tunnel. Traffic from inside machines to the remote nodes is similarly protected. What is distinct is that traffic from remote nodes to the rest of the Internet is governed by the central site's security policy. That is, the firewall administrator distributes a security policy to the remote nodes, as described above. Ideally, this same policy statement is used to control the traditional firewall, thus ensuring a consistent security policy.

Another variety of hybrid implementation ignores IPSEC entirely. In this situation, address-dependent policy rules are distributed to, and enforced by, every individual host within a site. Many newer systems support such functionality in the kernel. While address-based authentication is quite weak, if a simple router prevents

address-spoofing from the outside the security should comparable to that of traditional firewalls. Here, we use system management techniques to ensure consistent policy. We also rely on topology, thus forfeiting the ability to protect remote hosts. However, we still eliminate the single chokepoint and point of failure. A final hybrid scheme

5 combines the two previous hybrid schemes. Again, a simple router prevents address-spoofing by outside machines that talk to inside nodes. IPSEC is used to tunnel traffic from inside notes to remote nodes. On these systems, the IPSEC module provides anti-spoofing protection. Finally, all protected machines, whether local or remote, receive and enforce an address-based firewall policy.

10 With the proper sort of routing, triangle routing can be used for a few protocols, while distributed firewall techniques are used for most. Suppose, for example, that some protocol requires an application-level proxy for proper firewalling. Packets for that protocol can be routed through an IPSEC-protected tunnel to the inside of the firewall. From there, they can be forwarded to the Internet,

15 but only after proper processing by the firewall. A policy configuration file to implement such a scheme is as follows:

```
{Start}
    inside_net = x509{name="*.example.com", ver>19990315};
20 mail_gw = x509{name="mailgw.example.com"};
    time_server = IPv4{10.1.2.3};

    allow smtp(*, mail_gw);
    allow smtp(mail_gw, inside_net);
25 allow ntp(time_server,inside_net);
    allow *(inside_net, *);
{End}
```


Although various embodiments are specifically illustrated and described herein, it will be appreciated that modifications and variations of the present invention are covered by the above teachings and within the purview of the appended claims without departing from the spirit and intended scope of the invention. In particular,

5 one can use other key exchange techniques that don't use certificates, but instead use other host identifiers. For another example, although a public key cryptography algorithm was used in the embodiments of the invention, it can be appreciated that any number of encryption algorithms can be used to secure an address list and still fall within the scope of the invention. In another example, although specific equipment

10 was illustrated in FIG. 2 for a particular type of medium access technique, it can be appreciated that the packet filter processor shown in FIG. 2 can be modified to include equipment for any type of medium access technique, such as IEEE 802.2, 802.4, 802.5, 802.12 and so forth, and still fall within the scope of the invention.

What is claimed is:

- 1 1. A method for filtering packets, comprising:
2 receiving a packet sent from a first device to a second device;
3 authenticating an identifier for said packet;
4 determining whether to send said packet to said second device; and
5 sending said packet to said second device in accordance with said
6 determination.
- 1 2. The method of claim 1, wherein said determining comprises:
2 comparing said identifier to a list of identifiers;
3 retrieving at least one policy rule;
4 determining whether to send said packet to said second device in accordance
5 with said comparison and said policy rule.
- 1 3. The method of claim 1, wherein said identifier is a common host identifier.
- 1 4. The method of claim 1, wherein said authenticating is performed in accordance
2 with IPSEC standards.
- 1 5. The method of claim 1, wherein said authenticating comprises:
2 retrieving a pointer to a security association from an authentication header
3 from said packet;
4 retrieving a key associated with said security association; and
5 determining whether said packet is authentic using said key.

1 6. The method of claim 5, wherein said identifier is not authentic, further
2 comprising sending a first message to a third device indicating said identifier is not
3 authentic.

1 7. The method of claim 5, wherein said authentication header is an IPSEC
2 authentication header.

1 8. The method of claim 1, wherein said packet is encrypted prior to said
2 receiving, further comprising decrypting said packet prior to authenticating.

1 9. The method of claim 8, wherein said packet is encrypted and decrypted using
2 one of group of cryptographic techniques comprising DES, triple DES, HMAC and
3 RSA.

1 10. The method of claim 1, wherein said policy rule is stored in a policy
2 configuration file at said second device.

1 11. A machine-readable memory whose contents cause a computer system to
2 perform packet filtering, by performing the steps of:
3 receiving a packet sent from a first device to a second device;
4 authenticating an identifier for said packet;
5 determining whether to send said packet to said second device; and
6 sending said packet to said second device in accordance with said
7 determination.

1 12. The machine-readable memory of claim 11, wherein said determining
2 comprises:
3 comparing said identifier to a list of identifiers;
4 retrieving at least one policy rule;
5 determining whether to send said packet to said second device in accordance
6 with said comparison and said policy rule.

1 13. The machine-readable memory of claim 11, wherein said identifier is a
2 common host identifier.

1 14. The machine-readable memory of claim 11, wherein said authenticating is
2 performed in accordance with IPSEC standards.

1 15. The machine-readable memory of claim 11, wherein said authenticating
2 comprises:
3 retrieving a pointer to a security association from an authentication header
4 from said packet;
5 retrieving a key associated with said security association; and
6 determining whether said packet is authentic using said key.

1 16. The machine-readable memory of claim 15, wherein said identifier is not
2 authentic, further comprising sending a first message to a third device indicating said
3 identifier is not authentic.

1 17. The machine-readable memory of claim 15, wherein said authentication header
2 is an IPSEC authentication header.

1 18. The machine-readable memory of claim 11, wherein said packet is encrypted
2 prior to said receiving, further comprising decrypting said packet prior to
3 authenticating.

1 19. The machine-readable memory of claim 18, encrypted and decrypted using one
2 of group of cryptographic techniques comprising DES, triple DES, HMAC and RSA.

1 20. The machine-readable memory of claim 11, wherein said policy rule is stored
2 in a policy configuration file at said second device.

1 21. A packet filter for a distributed firewall, comprising:
 2 an input means coupled to said first network for receiving a data packet from a
 3 first device, said data packet having an encrypted common host identifier;
 4 a first buffer coupled to said input means for storing said received packet;
 5 a first memory segment containing a list of common host identifiers and at
 6 least one policy rule;
 7 a second memory segment for storing a program for decrypting said common
 8 host identifier, authenticating said common host identifier, and determining whether to
 9 send said packet to a second device based on said list and said policy rule;
 10 a processor coupled to said first buffer, said first memory segment and said
 11 second memory segment for executing said program; and
 12 an output means coupled to said first buffer for forwarding said compared data
 13 packet to said second device based on said comparison.

1 22. The apparatus of claim 1, further comprising a second buffer for storing said
 2 compared data packet prior to forwarding said compared data packet to the second
 3 device.

1 23. The apparatus of claim 22, wherein said random access memory comprises
 2 dynamic random access memory.

1 24. The apparatus of claim 23, further comprising a non-volatile random access
 2 memory for storing parameters used by said operating system program.

1 25. The apparatus of claim 24, further comprising means for receiving an updated
 2 list of origination addresses.

1 26. The apparatus of claim 25, wherein said means for receiving comprises an
2 asynchronous terminal device and a serial port coupled to said dynamic random access
3 memory.

1 27. The apparatus of claim 25, wherein said means for receiving comprises a
2 network interface card coupled to said dynamic random access memory.

1 28. The apparatus of claim 21, wherein said first network is a wireless network,
2 and said input means comprises means for receiving said data packets from said
3 wireless network.

1 29. A distributed firewall system, comprising:
2 a first network device;
3 a second network device in communication with said first network device;
4 a packet filter processor for each network device;
5 an encryption means coupled to said packet filter processor, said encryption
6 means for decrypting and authenticating a packet sent between said first network
7 device and said second network device; and
8 a system management module to manage said packet filter processors.

ABSTRACT OF DISCLOSURE:

- 1 A method and apparatus for a implementing a distributed firewall is described.
- 2 A packet filter processor receives a packet sent from a first device to a second device.
- 3 The packet filter processor authenticates an identifier for the packet. For example,
- 4 authentication could be performed using a cryptographically-verifiable identifier. The
- 5 packet filter processor determines whether to send the packet to the second device,
- 6 based on the authentication and a set of policy rules. The packet filter processor sends
- 7 the packet to the second device in accordance with the determination.

11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
219

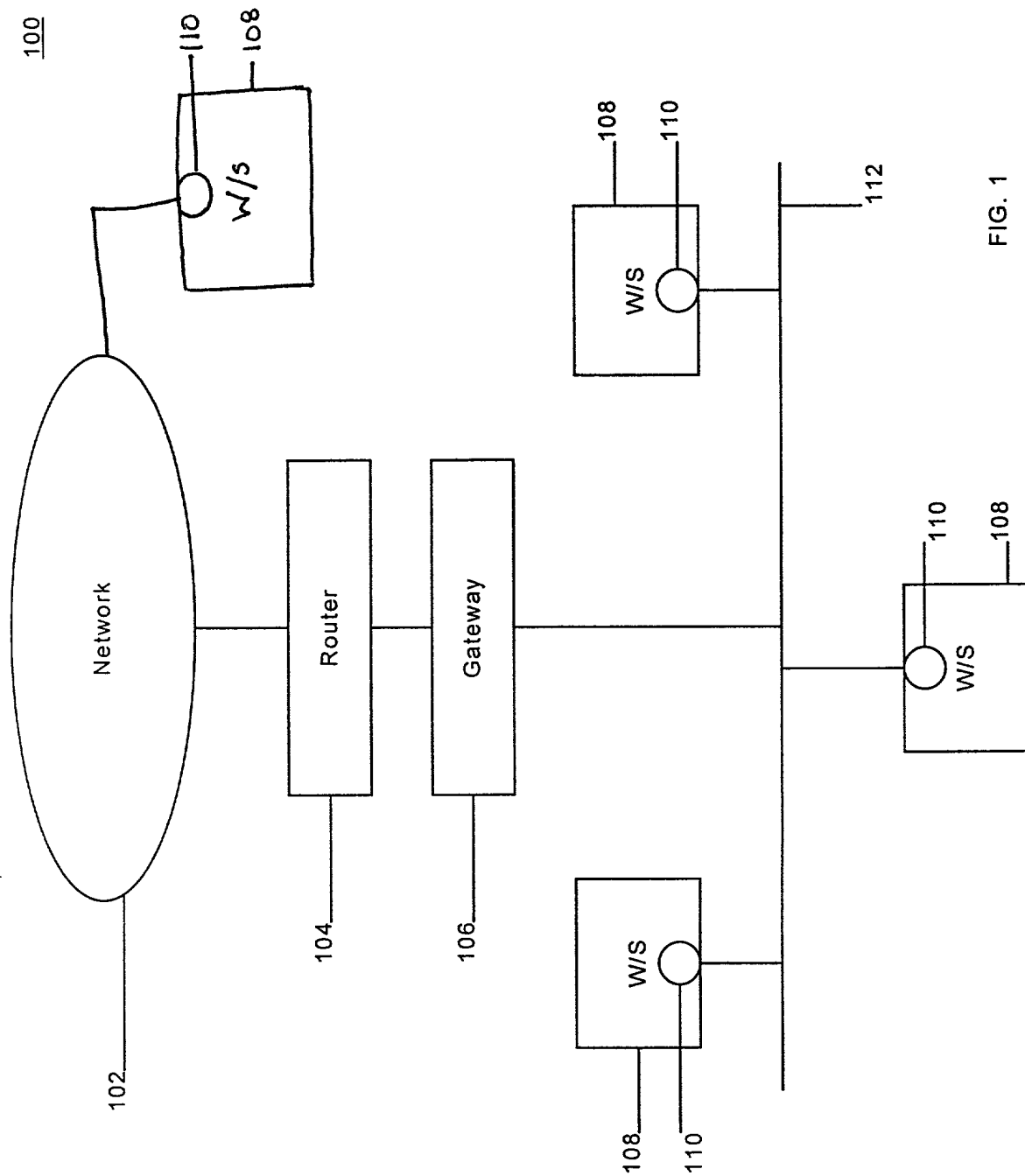


FIG. 1

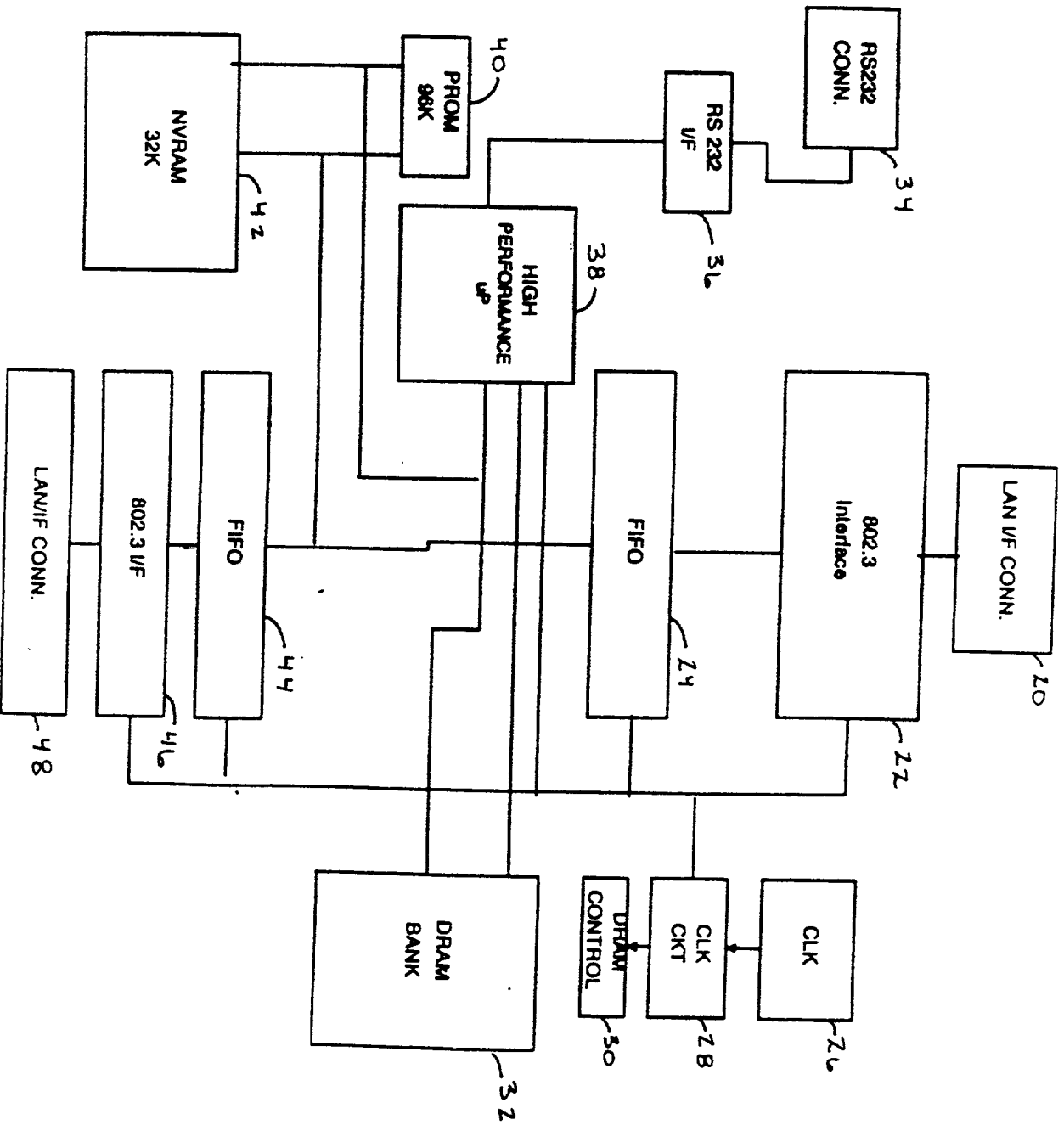


FIG. 2

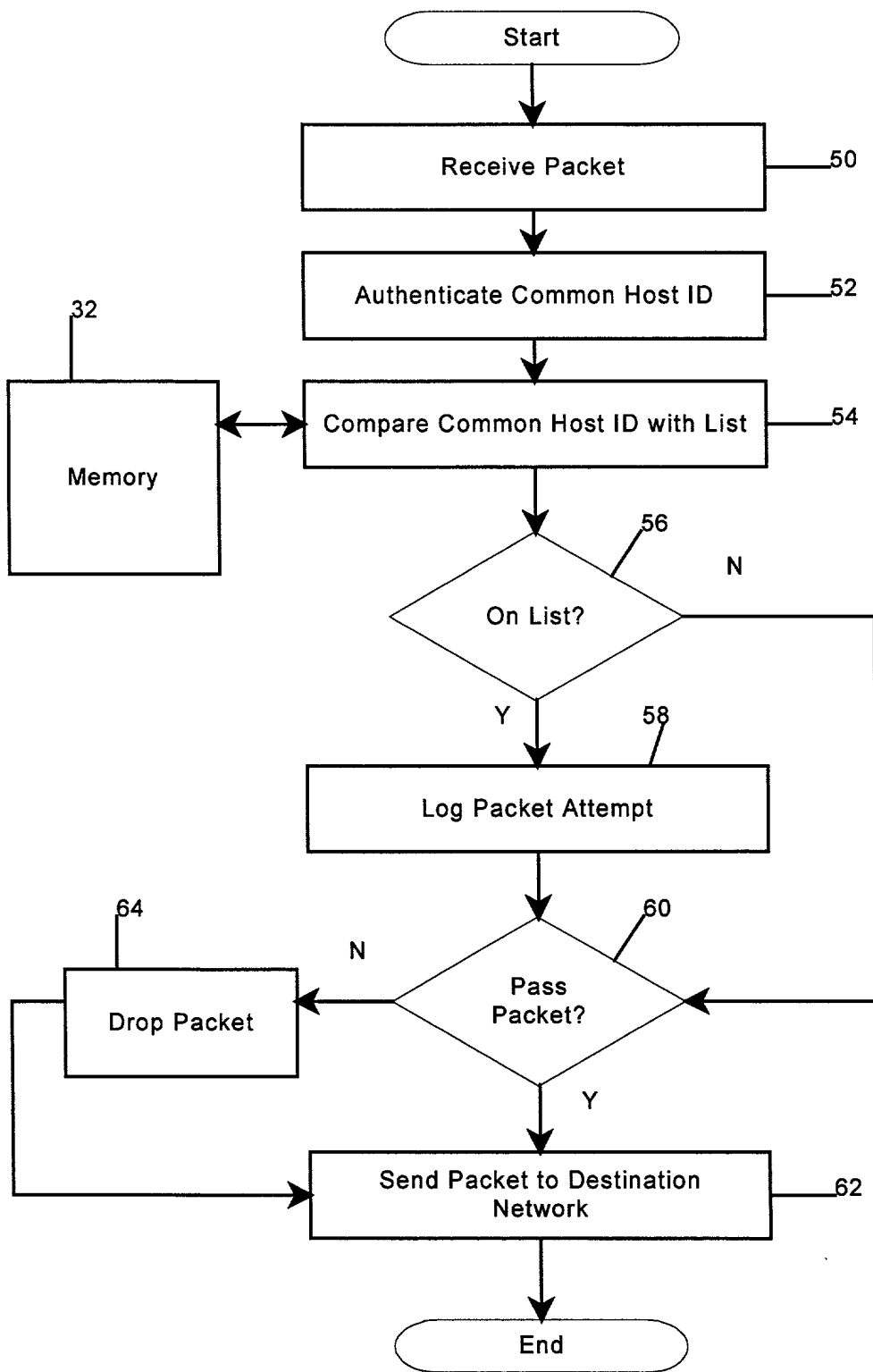


FIG. 3